The Consultative Committee for Space Data Systems

Draft Recommendation for
Space Data System Standards

SPACE LINK EXTENSION—RETURN ALL FRAMES SERVICE SPECIFICATION

DRAFT RECOMMENDED STANDARD

CCSDS 911.1-P-2.1

PINK SHEETS

December 2008

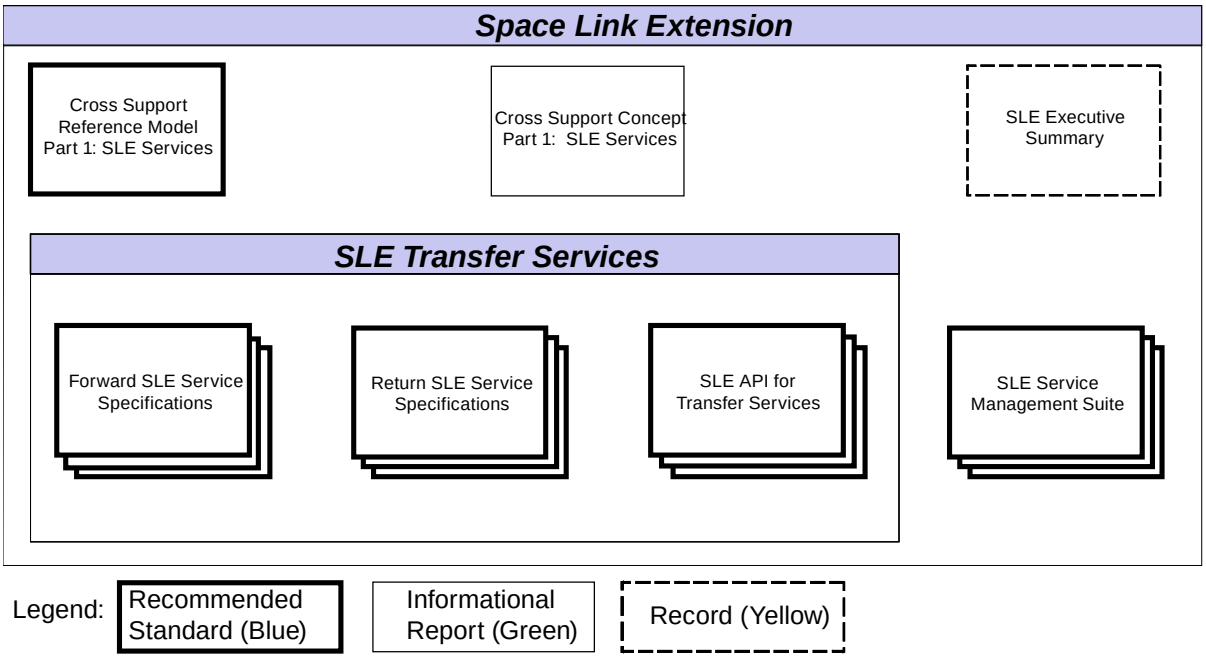

Figure 1-1: SLE Services Documentation

a) *Cross Support Concept—Part 1: Space Link Extension Services* (reference [E2]): a Report introducing the concepts of cross support and the SLE services;

b) *Cross Support Reference Model—Part 1: Space Link Extension Services* (reference [1]): a Recommended Standard that defines the framework and terminology for the specification of SLE services;

c) *SLE Return Service Specifications*: a set of Recommended Standards that will provide specification of all return link SLE services (this Recommended Standard is one of the specifications in that set);

d) *SLE Forward Service Specifications*: a set of Recommended Standards that will provide specification of all forward link SLE services;

e) *SLE API for Transfer Services Specifications*: a set of Recommended Standards that provide specifications of an Application Program Interface and a mapping to TCP/IP as underlying communications service for SLE services;

f) *SLE Service Management Specifications*: a set of Recommended Standards that establish the basis of SLE service management.

1.6.1.3 Definitions from TM Synchronization and Channel Coding

This Recommended Standard makes use of the following terms defined in reference [2]:

a) Attached Sync Marker;

b) codeblock;

c) convolutional code;

d) pseudo-randomization;

e) Reed-Solomon check symbols;

f) Reed-Solomon code;

g) turbo code.

1.6.1.4 Definitions from TM Space Data Link Protocol

This Recommended Standard makes use of the following term defined in reference [3]:

a) Frame Error Control Field (FECF);

b) TM Transfer Frame.

1.6.1.5 Definitions from AOS Space Data Link Protocol

This Recommended Standard makes use of the following terms defined in reference [4]:

a) Cyclic Redundancy Code (CRC);

b) AOS Transfer Frame;

c) Frame Error Control Field (FECF).

1.6.1.6 Definitions from SLE Reference Model

This Recommended Standard makes use of the following terms defined in reference [1]:

a) abstract binding;

b) abstract object;

c) abstract port;

d) abstract service;

e) invoker;

f) Mission Data Operation System (MDOS);

g) Mission User Entity (MUE);

h) offline delivery mode;

i) online delivery mode;

j) operation;

k) performer;

l) physical channel;

m) return data;

n) Return All Frames channel (RAF channel);

o) Return All Frames service (RAF service);

p) service agreement;

q) service provider (provider);

r) service user (user);

s) SLE Complex;

t) SLE Complex Management;

u) SLE data channel;

v) SLE Functional Group (SLE-FG);

w) SLE Protocol Data Unit (SLE-PDU);

x) SLE Service Data Unit (SLE-SDU);

y) SLE service package;

~~z) SLE System;~~

z) SLE transfer service instance;

aa) SLE transfer service production;

bb) SLE transfer service provision;

cc) SLE Utilization Management;

dd) space link;

ee) space link data channel;

ff) Space Link Data Unit (SL-DU);

gg) space link session.

1.6.1.7.6 Initiator

The initiator is the object that issues the request to bind to another object (the responder).

NOTE – In other words, the initiator is always the invoker of the request to bind to another object. Therefore, in the context of the request to bind, the terms 'initiator' and 'invoker' refer to the same object and are synonyms.

1.6.1.7.7 Invocation

The invocation of an operation is the making of a request by an object (the invoker) to another object (the performer) to carry out the operation.

1.6.1.7.8 Parameter

A parameter of an operation is data that may accompany the operation's invocation or return.

NOTE – The term parameter is also used to refer to mission-dependent configuration information used in the production or provision of the service.

1.6.1.7.9 Performance

The performance of an operation is the carrying out of the operation by an object (the performer).

1.6.1.7.10 Port Identifier

A port identifier identifies a source or a destination in a communications system.

NOTE – See 2.6.4.5 for more information.

1.6.1.7.11 Responder

The responder is the object that receives a request to bind and completes the binding (if possible) with the initiator in order for a service association to exist between the two objects.

NOTE – In other words, the responder is always the performer of the binding. Therefore, in the context of binding, the terms 'responder' and 'performer' refer to the same object and are synonyms.

2.4 ARCHITECTURE MODEL—FUNCTIONAL VIEW

2.4.1 RETURN SPACE LINK PROCESSING FUNCTIONAL GROUP

The Return Space Link Processing Functional Group (shown in figure 2-1) is the SLE functional group (SLE-FG) that produces the RAF service and the Return Insert service. The latter service is not addressed by this Recommended Standard but is defined in reference [1].

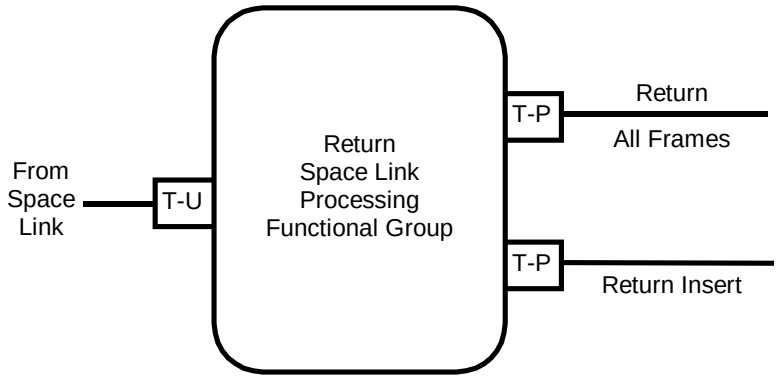

Figure 2-1: Return Space Link Processing SLE-FG

As described in reference [1], the Return Space Link Processing Functional Group consumes a space link data channel consisting of a stream of space link data units (SL-DUs). It produces an SLE data channel (viz., an RAF channel) consisting of a stream of SLE service data units (SLE-SDUs) for delivery to users. The SL-DUs are the telemetry frames carried on the space link physical channel. Most SLE-SDUs encapsulate an SL-DU; such SLE-SDUs also carry annotation information associated with that SL-DU (e.g., its ERT). Other SLE-SDUs carry notifications of the occurrences of certain events while the RAF service is being produced (e.g., loss of frame synchronization). More specifically, the Return Space Link Processing Functional Group performs the following functions with respect to RAF service:

a) utilizes the underlying antenna steering, microwave amplification, and receiver tuning capabilities of the ground element to acquire a radio frequency (RF) carrier channel;

b) demodulates a physical channel, consisting of a stream of digital symbols, from the RF carrier channel;

c) synchronizes and recovers telemetry frames from the physical channel;

d) performs convolutional decoding, removal of pseudo-randomization, ~~error control field~~Frame Error Control Field decoding, and/or Reed-Solomon or Turbo decoding as applicable (see references [2], [3], and [4]);

e) annotates each frame to form an RAF SLE-SDU and injects the resulting RAF SLE-SDU into the RAF channel associated with the physical channel from which the frame was extracted;

f) optionally, stores (and subsequently retrieves) sufficient data to reconstruct the RAF channel for delivery through one or more offline RAF service instances;

g) makes the RAF channel available to online and offline RAF service instances to enable the provision of RAF service.

The telemetry frames delivered by the RAF service are encapsulated in SLE-SDUs. RAF SLE-SDUs that encapsulate telemetry frames are annotated with information that pertains to that specific frame. The annotation consists of:

a) the ERT of the frame;

b) an identifier that indicates the antenna used to acquire the frame;

c) an indication of whether the frame contains detected, uncorrectable bit errors;

d) a parameter that characterizes the data link continuity of this frame with respect to the preceding frame on the space link;

e) an optional octet string that may be used to provide additional, non-standard annotations that are mutually agreed to by the SLE Complex providing the service and the Mission Data Operations System (MDOS) associated with the user of the service.

In the case that an acquired frame is Reed-Solomon encoded and the frame quality is 'good' (i.e., the frame is decoded successfully), then the Reed-Solomon check symbols are removed from the frame prior to its encapsulation into the SLE-SDU. If the frame quality of a Reed-Solomon encoded frame is 'erred' (i.e., the frame is undecodable), then the entire frame, including the Reed-Solomon check symbols, is encapsulated into the SLE-SDU for delivery to the user.

NOTES

1 SLE Return Space Link Processing requires that, at any given time, the coding options must be the same for all frames on a physical channel. This is more restrictive than the constraints imposed by references [2], [3], and [4]. In particular, Reed-Solomon coding must be present or absent on all of the frames of a physical channel. RAF service is not supported where there is a concurrent mix, on one physical channel, of some frames with Reed-Solomon coding and some frames without. Similarly, turbo coding must be present or absent on all of the frames of a physical channel. RAF service is not supported where there is a concurrent mix, on one physical channel, of some frames with turbo coding and some frames without.

involved in the exchange of SLE-PDUs are generally minor. However, the way an association is established (i.e., the binding) tends to vary significantly depending on the communications technology in use. Nonetheless, the RAF-BIND and RAF-UNBIND operations as specified in this document are intended to be 'technology neutral'. This neutrality is achieved as described in the following subsections.

For purposes of the communications mapping, the endpoints of an SLE association are identified by port identifiers, namely, an 'initiator port identifier' and a 'responder port identifier'. The port identifiers represent all the technology-specific addressing information needed to establish communications between the user and provider and to route SLE-PDUs between them. The initiator port identifier identifies the endpoint that will invoke the RAF-BIND operation (initiator). The responder port identifier identifies the endpoint that will perform the RAF-BIND operation (responder). Generally speaking, the information represented by a port identifier consists of:

a) information needed in order to route data between two real systems over a communications channel or network; and

b) information needed in order to route data within a real system to a particular application entity.

For example, the information represented by a port identifier might be the combination of an Internet Protocol (IP) network address and a Transmission Control Protocol (TCP) port number or the combination of an OSI network address and an associated set of Service Access Points (SAPs).

The exact relationship between SLE port identifiers and communications ports provided by the underlying communications service must be specified by the mapping of the RAF service to the underlying communications service. If the underlying communications service is connection-oriented, then the mapping may specify a one-to-one relationship between SLE associations and communications connections; however, that is not required. For example, two SLE associations involving the same pair of SLE endpoints may share a single connection. In that case, it is the responsibility of the mapping of the RAF service to the underlying communications service to specify how the SLE-PDUs of one association are distinguished from the SLE-PDUs of the other association.

One possible mapping of the SLE transfer service to the TCP/IP communications service is specified in [E6]. As part of this mapping, also issues such as sizing of TCP buffers in accordance with the bandwidth-delay product of the communication link and ways to manage relative priority of transfer services concurrently using the same connectivity are to be addressed.

In order for an SLE association to be established, SLE Complex Management and SLE Utilization Management must agree beforehand on the responder port identifier for the association. The responder needs the information represented by the responder port identifier to ensure that resources are allocated to recognize and respond to an RAF-BIND

3.1.5 AUTHENTICATION

NOTE – Requirements for security depend on the application and the ~~SLE system~~ environment of the SLE Complexes and the MDOS (e.g., whether closed or public networks are used or if access is only from physically restricted areas). In many environments, security may be provided by the communications service, transparently to the SLE application. This Recommended Standard does not preclude the use of security features that are provided by the communications service or the local environment, nor does it assume the availability of such features.

3.1.5.1 The RAF service shall provide the following options with respect to the level of authentication of invocations and returns of operations:

a) 'all': all RAF invocations and returns, except the invocation of RAF-PEER-ABORT, shall be authenticated;

b) 'bind': only the RAF-BIND invocation and return shall be authenticated;

c) 'none': no RAF invocations or returns shall be authenticated.

3.1.5.2 SLE Complex Management and SLE Utilization Management shall agree on the level of authentication to be required for an association between a service user and a service provider and shall configure both entities accordingly.

3.1.5.3 SLE Complex Management and SLE Utilization Management shall agree on the algorithm used to generate and check credentials parameters and shall make this algorithm known to the service user and service provider together with associated parameters such as passwords or keys as necessary for the selected algorithm.

NOTES

1 The specification of the algorithms themselves is outside the scope of this Recommended Standard.

2 The `initiator-identifier` and `responder-identifier` parameters of the RAF-BIND operation identify the user and provider and therefore the applicable authentication level and algorithm necessary to generate and check credentials.

3.1.5.4 For operations for which authentication is required by the terms of the agreement between SLE Complex Management and SLE Utilization Management:

a) invocations shall include an `invoker-credentials` parameter to permit the performer to authenticate the invocation;

b) returns shall include a `performer-credentials` parameter to permit the invoker to authenticate the return.

3.1.6.8 Compliance with this Recommended Standard does not require the performer to process invocations concurrently; however, the performer must accept invocations from a non-blocking invoker and buffer and serialize them by local means not visible externally.

3.1.7 TIME

3.1.7.1 The time reference for all parameters containing a time value shall be based on Coordinated Universal Time (UTC).

NOTE – The type of all time parameters is specified in annex A.

3.1.7.2 The `earth-receive-time` parameter (see 3.6.2.3) shall be expressed using the CCSDS Day Segmented (CDS) time code (reference [5]) with ~~a resolution of one microsecond,~~ an epoch of 1958-01-01, and a 16-bit day segment. Depending on the RAF service provider capabilities and/or the supported mission requirements, the time tag may have either a resolution of microseconds or a resolution of picoseconds.

3.1.7.3 The `earth-receive-time` parameter shall have a precision of one millisecond or better.

3.1.7.4 The `earth-receive-time` parameter shall be accurate to within one millisecond or better.

3.1.8 SETTING OF PARAMETERS

An RAF provider shall permit setting of the service configuration parameters as specified in table 3-1.

The range or set of values a parameter may assume is constrained by specification of its data type (see annex A).

Service management may further constrain the allowed values for a given service instance.

Table 3-1: Setting of RAF Service Configuration Parameters

Parameter	Service Management	RAF-START Operation	RAF-SCHEDULE-STATUS-REPORT Operation
delivery-mode	X		
latency-limit	X		
maximum-delivery-rate	X		
maximum-reporting-cycle	X		
minimum-reporting-cycle	X		
permitted-frame-quality-set	X		
reporting-cycle			X
requested-frame-quality		X	
return-timeout-period	X		
service-instance-provision-period	X		
service-version-number	X		
transfer-buffer-size	X		

NOTES

1 The user can ascertain the current value of the parameters presented in table 3-11 by means of the RAF-GET-PARAMETER operation.

2 This Recommended Standard also refers to parameters that are set by service management, but are not listed in table 3-1. These parameters cannot be ascertained by means of the RAF-GET-PARAMETER operation.

3 The methods used by service management to control service provision and service production parameters are outside the scope of this Recommended Standard.

3.1.9 DELIVERY MODES

3.1.9.1 Timely Online Delivery Mode

3.1.9.1.1 For timely online delivery mode, the RAF service provider shall store frames acquired from the space link and certain information associated with those frames (as per 3.6.2) in a buffer called the transfer buffer. The stored information shall be an RAF-TRANSFER-DATA invocation or the equivalent thereof.

3.2.2.8 `version-number`

3.2.2.8.1 The **`version-number`** parameter shall identify the version number of the RAF service specification that is to govern this association if RAF-BIND succeeds.

3.2.2.8.2 `version-number` is conditionally present in the return based on the `result` parameter:

a) if the value of `result` is 'positive result', `version-number` shall be present in the return;

b) if the value of `result` is 'negative result', `version-number` shall not be present in the return.

3.2.2.8.3 If the value of the `result` parameter is 'positive result', the responder shall either:

a) accept the version proposed by the initiator by putting the same version number into the return; or,

b) if the responder supports version negotiation, propose a lower (earlier) version number by putting the lower number into the return.

3.2.2.8.4 If the responder implementation does not support the requested version and does not support a lower version (or does not support version negotiation), the responder shall reject the bind with the `diagnostic` parameter set to 'version not supported'.

3.2.2.8.5 If the responder proposes a lower version and the initiator implementation does not support version negotiation or does not support the version proposed by the responder, the initiator shall unbind the association.

3.2.2.8.6 The value of the `version-number` parameter for the RAF service defined by this issue of this Recommended Standard shall be '~~2~~3'.

NOTE – The version negotiation process as outlined above is feasible only as long as future versions of the RAF service do not modify the specification of the RAF-BIND operation.

3.2.2.9 `service-instance-identifier`

The **`service-instance-identifier`** parameter shall uniquely identify this service instance within the scope of the service-providing SLE Complex.

3.6 RAF-TRANSFER-DATA

3.6.1 PURPOSE

3.6.1.1 The provider shall invoke the RAF-TRANSFER-DATA operation to deliver a telemetry frame to the user.

3.6.1.2 The RAF-TRANSFER-DATA operation shall be an unconfirmed operation.

NOTE – Although RAF-TRANSFER-DATA is an unconfirmed operation, it is assumed that the communications service provides certain guarantees, as described in 1.3.1.

3.6.1.3 RAF-TRANSFER-DATA is valid only in state 3 ('active') and shall be invoked only by the provider.

3.6.2 INVOCATION~~, RETURN,~~ AND PARAMETERS

3.6.2.1 General

The parameters of the RAF-TRANSFER-DATA operation shall be present in the invocation as specified in table 3-6.

Table 3-6: RAF-TRANSFER-DATA Parameters

Parameters	Invocation
`invoker-credentials`	M
`earth-receive-time`	M
`antenna-ID`	M
`data-link-continuity`	M
`delivered-frame-quality`	M
`private-annotation`	M
`data`	M

3.6.2.2 `invoker-credentials`

The **`invoker-credentials`** parameter shall provide information that enables the user to authenticate the RAF-TRANSFER-DATA invocation (see 3.1.5).

3.6.2.3 `earth-receive-time`

The **`earth-receive-time`** parameter shall contain the UTC time at which the signal event corresponding to the leading edge of the first symbol which has been influenced by the last bit of the attached sync marker that immediately preceded this telemetry frame was presented at the phase center of the antenna used to acquire the frame.

~~NOTE The first bit of the frame is the first bit following the attached sync marker.~~

NOTE – In case of punctured coding, the number of symbols influenced by each information bit is variable, depending on the puncture pattern. To minimize the resulting jitter of the `earth-receive-time` annotation with respect to the beginning of the frame, the end of the attached sync marker is used as the reference event.

3.6.2.4 `antenna-ID`

3.6.2.4.1 The **`antenna-ID`** parameter shall indicate which antenna of the SLE Complex was used to acquire this frame.

NOTE – `antenna-ID` is provided specifically to identify the physical location used as the reference point for the `earth-receive-time` parameter.

3.6.2.4.2 SLE Complex Management and SLE Utilization Management shall mutually agree upon the allowable values for `antenna-ID` and their interpretation.

NOTE – It is assumed that the value of the `antenna-ID` parameter is a reference to the actual location information, which is provided outside the scope of this service.

3.6.2.5 `data-link-continuity`

The **`data-link-continuity`** parameter shall indicate whether this frame was the direct successor of the previous frame on the space link selected by means the RAF-START operation.

NOTE – Even though this frame may have been the direct successor of the previous frame acquired from the space link, it may not be the direct successor of the previous frame delivered to the user of this service instance (e.g., because of the setting of `requested-frame-quality`, see 3.4.2.7).

3.6.2.5.1 The **`data-link-continuity`** parameter shall contain an integer value:

a) a value of '–1' shall indicate that this is the first frame after the start of production;

b) a value of '0' shall indicate that this frame is the direct successor to the last frame acquired from the space link by RAF production;

c) any non-zero positive value shall indicate that this frame is not the direct successor to the last frame acquired from the space link:

1) a non-zero positive value further indicates an estimate of the number of frames that were missed since the last frame acquired before this frame;

2) a value of '1' may be used if no better estimate is available.

3.6.2.6 `delivered-frame-quality`

3.6.2.6.1 The **`delivered-frame-quality`** parameter shall indicate the result of Reed-Solomon decoding or ~~error control field~~Frame Error Control Field decoding and shall contain one of the following values:

a) 'good'—for Reed-Solomon encoded frames, the frame, after decoding, consists entirely of codewords of the Reed-Solomon code; or, for frames that are not Reed-Solomon encoded, decoding of the ~~error control field~~Frame Error Control Field indicates that the frame does not contain any errors;

b) 'erred'—for Reed-Solomon encoded frames, the frame, after decoding, contains at least one codeword that is not a codeword of the Reed-Solomon code; or, for frames that are not Reed-Solomon encoded, decoding of the ~~error control field~~Frame Error Control Field indicates that the frame contains at least one error.

c) 'undetermined' – the results of the decoding process for this frame are not specified. It may be~~,~~ that decoding was not performed at all ~~but~~, that a non-standard decoding was performed, or that a decoding was performed but the results are not provided for whatever reason. The exact meaning of this value of the parameter is implementation dependent.

NOTE – The 'undetermined' `delivered-frame-quality` is only provided if the RAF-START operation specifies 'all' for the requested-frame-quality.

3.6.2.6.2 If a Reed-Solomon encoded frame is successfully decoded, it is assumed to be error free, and the ~~error control field~~Frame Error Control Field of such a frame may be ignored.

NOTES

1 Reed-Solomon coding is specified in reference [2]. Decoding of the ~~error control field~~Frame Error Control Field is specified in references [3] and [4].

2 As indicated in 1.3.2.2, provision of RAF service requires that, at any given time, the coding options must be the same for all frames on a physical channel, which is more restrictive than what is specified in references [2], [3] and [4].

3 Whether Reed-Solomon coding or turbo coding is in use or not is managed information; i.e., it is not signaled in the data itself but must be known a priori. It is assumed that this information is conveyed through service management.

3.6.2.7 `private-annotation`

The **`private-annotation`** parameter shall be used to convey additional information that may be associated with a frame:

a) it may be set to 'null' to indicate that there is no private annotation;

b) if not 'null', there must be a prior arrangement between SLE Complex Management and SLE Utilization Management regarding the contents and interpretation of this parameter.

3.6.2.8 `data`

The value of the **`data`** parameter shall be the telemetry frame acquired by the provider from the space link for delivery to the user.

~~a) if `delivered-frame-quality` is 'good', `data` is a transfer frame or a AOS transfer frame;~~

~~b) if `delivered-frame-quality` is 'erred', `data` is a data unit of the same length as a frame that, during frame synchronization, was identified as a frame but, on decoding, was found to contain errors.~~

a) if `delivered-frame-quality` is 'good' and the frame is not Reed-Solomon encoded, then `data` is a TM transfer frame or an AOS transfer frame;

b) if `delivered-frame-quality` is 'good' and the frame is Reed-Solomon encoded, then the Reed-Solomon check symbols are discarded, and the resulting TM transfer frame or AOS transfer frame is delivered as `data`;

c) if `delivered-frame-quality` is 'erred' and the frame is not Reed-Solomon encoded, then `data` is a data unit of the same length as a (if applicable, convolutionally or turbo decoded) frame that, during frame synchronization, was identified as a frame but, on FECF checking, was found to contain errors;

d) if `delivered-frame-quality` is 'erred' and the frame is Reed-Solomon encoded (i.e., if a candidate codeblock cannot be Reed-Solomon decoded), then the entire candidate codeblock as presented at the input to the Reed-Solomon decoder is delivered unmodified as `data`;

e) if `delivered-frame-quality` is 'undetermined', `data` is a data unit of the same length as a TM transfer frame or AOS transfer frame that during frame synchronization was identified as a candidate frame, but of which the coding scheme is unknown or not supported so that the frame quality cannot be determined by decoding.

NOTES

1 The specification in item e) above allows that the length of the candidate frame that is delivered (when delivered-frame-quality is 'undetermined') is implementation dependent; i.e., it may be a frame or codeblock in the symbol or bit domain.

~~1~~2 The value of the `data` parameter does not include the attached sync marker.

~~2 For Reed-Solomon encoded frames, if a codeblock or coded transfer frame is successfully Reed-Solomon decoded and the user has requested the delivery of 'all frames' or 'good frames only' (see 3.4.2.7), the Reed-Solomon check symbols are discarded, and the resulting transfer frame or AOS transfer frame is delivered with `delivered-frame-quality` set to 'good'. If a candidate codeblock or coded transfer frame cannot be Reed-Solomon decoded and the user has requested the delivery of 'all frames' or 'erred frames only', the entire candidate codeblock or coded transfer frame is delivered with `delivered-frame-quality` set to 'erred'.~~

3.6.3 EFFECTS

The RAF-TRANSFER-DATA operation shall have the following effects:

a) a telemetry frame acquired by the provider from the space link shall be delivered to the user;

b) the provider shall remain in state 3 ('active').

3.7 RAF-SYNC-NOTIFY

3.7.1 PURPOSE

3.7.1.1 The RAF service provider shall invoke the RAF-SYNC-NOTIFY operation to notify the user of the occurrence of an event affecting the production of the RAF service.

NOTE – Notification of events may be of value to the user in understanding specific provider behavior, such as an interruption in frame delivery.

3.7.1.2 The RAF-SYNC-NOTIFY operation shall be an unconfirmed operation.

3.7.1.3 The order in which the RAF-SYNC-NOTIFY and RAF-TRANSFER-DATA operations are invoked shall reflect the actual chronology of events.

NOTE – For example, if an RAF-SYNC-NOTIFY operation is invoked after one RAF-TRANSFER-DATA operation but before another, then the event indicated by the notification occurred after the ERT of the frame associated with the preceding RAF-TRANSFER-DATA but before the ERT of the frame associated with the following RAF-TRANSFER-DATA.

3.7.1.4 RAF-SYNC-NOTIFY is valid only in state 3 (‘active’) and shall be invoked only by the provider.

3.7.2 INVOCATION~~, RETURN,~~ AND PARAMETERS

3.7.2.1 General

The parameters of the RAF-SYNC-NOTIFY operation shall be present in the invocation as specified in table 3-7.

Table 3-7: RAF-SYNC-NOTIFY Parameters

Parameter	Invocation
`invoker-credentials`	M
`notification-type`	M
`notification-value`	C

3.7.2.2 `invoker-credentials`

The **`invoker-credentials`** parameter shall provide information that enables the user to authenticate the RAF-SYNC-NOTIFY invocation (see 3.1.5).

3.9 RAF-STATUS-REPORT

3.9.1 PURPOSE

3.9.1.1 The provider shall invoke the RAF-STATUS-REPORT operation to send a status report to the user.

3.9.1.2 RAF-STATUS-REPORT shall be an unconfirmed operation.

3.9.1.3 Status reports shall be sent (or not sent) in accordance with user requests conveyed by means of the RAF-SCHEDULE-STATUS-REPORT operation (see 3.8).

3.9.1.4 The RAF-STATUS-REPORT operation is valid only in states 2 (‘ready’) and 3 (‘active’) and shall be invoked only by the provider.

3.9.2 INVOCATION~~, RETURN,~~ AND PARAMETERS

3.9.2.1 General

The parameters of the RAF-STATUS-REPORT operation shall be present in the invocation as specified in table 3-9.

Table 3-9: RAF-STATUS-REPORT Parameters

Parameters	Invocation
`invoker-credentials`	M
`number-of-error-free-frames-delivered`	M
`number-of-frames-delivered`	M
`frame-sync-lock-status`	M
`symbol-sync-lock-status`	M
`subcarrier-lock-status`	M
`carrier-lock-status`	M
`production-status`	M

3.9.2.2 `invoker-credentials`

The **`invoker-credentials`** parameter shall provide information that enables the performer to authenticate the RAF-STATUS-REPORT invocation (see 3.1.5).

b) ‘negative result’—the RAF-GET-PARAMETER operation has not been performed for the reason specified in the `diagnostic` parameter.

Table 3-11: RAF Parameters

Parameter	Description
delivery-mode	The delivery mode for this instance of RAF service, which is set by service management (see 3.1.9): its value shall be ‘timely online delivery mode’, ‘complete online delivery mode’, or ‘offline delivery mode’
latency-limit	The maximum allowable delivery latency time (in seconds) for the online delivery mode, as defined in 3.1.9.1 (i.e., the maximum delay from when the frame is acquired by the provider until the RAF extracted from it is delivered to the user): the value of this parameter shall be ‘null’ if the delivery mode is offline.
reporting-cycle	The current setting of the reporting cycle for status reports (see 3.8 and 3.9): the value is ‘null’ if cyclic reporting is off, otherwise it is the time (in seconds) between successive RAF-STATUS-REPORT invocations (see 3.8).
requested-frame-quality	The frame quality criteria, set by the RAF-START operation, used to determine which frames are selected for delivery: if the provider is in state 3 (‘active’), its value shall be ‘good frames only’, ‘erred frames only’, or ‘all frames’; otherwise, its value shall be ‘undefined’;
return-timeout-period	The maximum time period (in seconds) permitted from when a confirmed RAF operation is invoked until the return is received by the invoker (see 4.1.3).
transfer-buffer-size	The size of the transfer buffer ~~(see 3.1.9)~~: the value of this parameter shall indicate the number of RAF-TRANSFER-DATA and RAF-SYNC-NOTIFY invocations that can be stored in the transfer buffer. The precise specification of the transfer buffer size may be found in 3.1.9.

3.10.2.8 `diagnostic`

3.10.2.8.1 If `result` is ‘negative result’, the **`diagnostic`** parameter shall be present in the return, and its value shall be one of the following:

a) ‘duplicate Invoke-ID’—the value of the `invoke-ID` parameter is the same as the `invoke-ID` of a previous, outstanding operation;

b) ‘unknown parameter’—the value of `raf-parameter` does not identify an RAF parameter that is recognized by the service provider;

3.11 RAF-PEER-ABORT

3.11.1 PURPOSE

3.11.1.1 The user or provider shall invoke the RAF-PEER-ABORT operations to notify the peer system that the local application detected an error that requires that the association between them be terminated abnormally.

3.11.1.2 RAF-PEER-REPORT shall be an unconfirmed operation.

3.11.1.3 RAF-PEER-ABORT is valid only in states 2 (‘ready’) and 3 (‘active’) and may be invoked by either the user or the provider.

3.11.2 INVOCATION~~, RETURN,~~ AND PARAMETERS

3.11.2.1 General

The parameters of the RAF-PEER-ABORT operation shall be present in the invocation as specified in table 3-12.

Table 3-12: RAF-PEER-ABORT Parameters

Parameters	Invocation
`diagnostic`	M

3.11.2.2 `diagnostic`

The **`diagnostic`** parameter shall specify why the RAF-PEER-ABORT is being invoked, and its value shall be one of the following:

a) ‘access denied’—a responder with an identity as presented in the `responder-identifier` parameter of the RAF-BIND return is not known to the initiator (e.g., the value of the `responder-identifier` parameter does not match the authorized responder for any service instance known to the initiator);

b) ‘unexpected responder ID’—the value of the `responder-identifier` parameter in the RAF-BIND return does not match the identity of the authorized responder for this service instance as specified by service management;

c) ‘operational requirement’—the local system had to terminate the association to accommodate some other operational need;

d) ‘protocol error’—the local application detected an error in the sequencing of RAF service operations;

4 RAF PROTOCOL

4.1 GENERIC PROTOCOL CHARACTERISTICS

NOTE – This section specifies the handling of invalid SLE-PDUs and other failures affecting the protocol.

4.1.1 UNEXPECTED PROTOCOL DATA UNIT

If the peer application invokes an operation not allowed in the current state of the performer, the performer shall abort the association by invoking the RAF-PEER-ABORT operation with the `diagnostic` parameter set to 'protocol error'.

4.1.2 INVALID PROTOCOL DATA UNIT

If the application receives an invocation or return that contains an unrecognized operation type, contains a parameter of the wrong type, or is otherwise not decodable, the application shall abort the association by invoking the RAF-PEER-ABORT operation with the `diagnostic` parameter set to 'encoding error'.

4.1.3 MISSING RETURN

For confirmed operations, if the invoker does not receive the return from the performer within a timeout period specified by service management, the invoker shall abort the association by invoking the RAF-PEER-ABORT operation with the `diagnostic` parameter set to 'return timeout'.

NOTES

1 The timeout period shall be chosen taking into account performance of user and provider applications as well as the delays introduced by the underlying communications service.

2 In order to provide responsive service and short timeout periods, the generation of the return from an operation must not depend on any human interaction.

3 After invoking the RAF-UNBIND operation, the initiator must not invoke any further operations with the exception of the case addressed in 3.3.1.4 nor send any returns. The responder is not required to send any pending returns after having received the RAF-UNBIND invocation. Therefore, following an RAF-UNBIND invocation, the 'missing return' event may occur.

Table 4-2: Event Description References

Event	Reference
'data available'	3.1.9.1.2, 3.1.9.2.2, 3.1.9.3.2
'end of data'	3.7.2.3
'end of service instance provision period'	3.11.2.2
'invalid protocol data unit'	4.1.2
'loss of frame synchronization'	3.7.2.3
'not authenticated SLE-PDU'	4.1.7
'production status change'	3.7.2.3
'release timer expired'	3.1.9.1.4, 3.1.9.2.6
'reporting-cycle timer expired'	3.8.2.6
'return SLE-PDU with unsolicited Invoke-ID'	4.1.4
'return <n> timer expired'	4.1.3
'start of service instance provision period'	1.6.1.7.13

Table 4-3: Predicate Descriptions

Predicate	Evaluates to TRUE if
"buffer empty"	There are no RAF SLE-PDUs in the transfer buffer
"buffer full"	The transfer buffer cannot accommodate the currently available annotated frame or synchronous notification
"compatible"	The version number contained in (+rafBindReturn) is supported by the provider
"complete online"	Delivery mode is complete online
~~"congested"~~	~~The underlying communications service cannot accept the contents of the transfer buffer because of congestion~~
"done"	The unbind-reason parameter value in the provider-initiated BIND invocation was 'end'
"end"	All checks on the UNBIND invocation are passed and the unbind-reason parameter value is 'end'
"immediately"	All parameter checks on the RAF-SCHEDULE-STATUS-REPORT are passed and the `report-request-type` value is 'immediately'
"offline"	Delivery mode is offline
"online"	Delivery mode is timely online or complete online
"periodically"	All parameter checks on the RAF-SCHEDULE-STATUS-REPORT are passed and the `report-request-type` value is 'periodically'
"positive result"	All checks on the invocation are passed
"provider initiated"	The RAF-BIND operation is specified to be initiated by the provider for this service instance
"provision period"	Current time is inside the service instance provision period
"retry permitted"	The diagnostic value contained in the (-rafBindReturn) is 'unable to comply' or 'other', and the service instance provision period is still active

Predicate	Evaluates to TRUE if
“timely online”	Delivery mode is timely online

Table 4-4: Boolean Flags

Flag Name	Initial Value
“bind pending”	FALSE
“congested”	FALSE
“unbind pending”	FALSE

Table 4-5: Compound Action Definitions

Name	Actions Performed
{clean up}	stop release timer stop all return timers stop reporting-cycle timer reinitialize transfer buffer reset parameter values to those specified in service package
{immediate report}	(rafStatusReportInvocation) stop reporting-cycle timer
{insert annotated frame}	annotate the available frame with the parameters of the RAF-TRANSFER-DATA operation insert the annotated frame into the transfer buffer
{invoke bind}	(rafBindInvocation) set “bind pending” to TRUE start return <n> timer
{invoke unbind}	(rafUnbindInvocation) stop reporting-cycle timer set “unbind pending” to TRUE start return <n> timer
{pass buffer contents}	stop release timer submit contents of transfer buffer to underlying communications service IF successful THEN set “congested” to FALSE ELSE set “congested” to TRUE reinitialize transfer buffer using the nominal size
{peer abort ‘xxxx’}	stop release timer stop all return timers stop reporting-cycle timer reinitialize transfer buffer (rafPeerAbortInvocation) with diagnostic set to ‘xxxx’

Name	Actions Performed
{periodic report}	(rafStatusReportInvocation) set reporting-cycle timer to the `reporting-cycle` value in the most recent SCHEDULE-STATUS-REPORT invocation start reporting-cycle timer
{provider unbind}	set “unbind pending” to FALSE stop all return timers
{return timeout}	(rafPeerAbortInvocation) with diagnostic ‘return timeout’ set “bind pending” to FALSE set “unbind pending” to FALSE
{start release timer}	set release timer to latency limit start release timer
{sync notify ‘xxxx’}	create an RAF synchronous notification with ~~diagnostic~~`notification-type` set to ‘xxxx’ insert the notification into the transfer buffer
{transmit buffer}	stop release timer Submit the contents of transfer buffer to underlying communications service until accepted by that service reinitialize transfer buffer
{user unbind}	stop reporting-cycle timer stop all return timers (rafUnbindReturn)

```
ParameterName                          ::=   INTEGER
{  apidList                                  (2)
,  bitLockRequired                           (3)
,  blockingTimeoutPeriod                     (0)
,  blockingUsage                             (1)
,  bufferSize                                (4)
,  deliveryMode                              (6)
,  directiveInvocation                       (7)
,  directiveInvocationOnline                 (108)
,  expectedDirectiveIdentification           (8)
,  expectedEventInvocationIdentification     (9)
,  expectedSlduIdentification                (10)
,  fopSlidingWindow                          (11)
,  fopState                                  (12)
,  latencyLimit                              (15)
,  mapList                                   (16)
,  mapMuxControl                             (17)
,  mapMuxScheme                              (18)
,  maximumFrameLength                        (19)
,  maximumPacketLength                       (20)
,  maximumSlduLength                         (21)
,  modulationFrequency                       (22)
,  modulationIndex                           (23)
,  permittedControlWordTypeSet               (101)
,  permittedGvcidSet                         (24)
,  permittedTcVcidSet                        (102)
,  permittedTransmissionMode                 (107)
,  permittedUpdateModeSet                    (103)
,  plopInEffect                              (25)
,  reportingCycle                            (26)
,  requestedControlWordType                  (104)
,  requestedFrameQuality                     (27)
,  requestedGvcid                            (28)
,  requestedTcVcid                           (105)
,  requestedUpdateMode                       (106)
,  returnTimeoutPeriod                       (29)
,  rfAvailable                               (30)
,  rfAvailableRequired                       (31)
,  segmentHeader                             (32)
,  subcarrierToBitRateRatio                  (34)
,  timeoutType                               (35)
,  timerInitial                              (36)
,  transmissionLimit                         (37)
,  transmitterFrameSequenceNumber            (38)
,  vcMuxControl                              (39)
,  vcMuxScheme                               (40)
,  virtualChannel                            (41)
}

SlduStatusNotification  ::=   INTEGER
{  produceNotification           (0)
,  doNotProduceNotification      (1)
}

SpaceLinkDataUnit       ::=   OCTET STRING (SIZE (41 .. 65536))
```

```
Time                    ::=   CHOICE
{  ccsdsFormat       [0]   TimeCCSDS
,  picoFormat        [1]   TimeCCSDSpico
}

TimeCCSDS               ::=   OCTET STRING (SIZE(8))
-- P-field is implicit (not present, defaulted to 41 hex
-- T-field:
-- 2 octets: number of days since 1958/01/01 00:00:00
-- 4 octets: number of milliseconds of the day
-- 2 octets: number of microseconds of the millisecond
--        (set to 0 if not used)
-- This definition reflects exactly the format of the CCSDS defined
-- time tag as used in spacelink data units (see Time Code Formats.
-- Recommendation for Space Data System Standards, CCSDS 301.0-B-3.
-- Blue Book.  Issue 3.  Washington, D.C.: CCSDS, January 2002).

TimeCCSDSpico           ::=   OCTET STRING (SIZE(10))
-- P-field is implicit (not present, defaulted to 42 hex
-- T-field:
-- 2 octets: number of days since 1958/01/01 00:00:00
-- 4 octets: number of milliseconds of the day
-- 4 octets: number of picoseconds of the millisecond
--        (set to 0 if not used)
-- This definition reflects exactly the format of the CCSDS defined
-- time tag as used in spacelink data units (see Time Code Formats.
-- Recommendation for Space Data System Standards, CCSDS 301.0-B-3.
-- Blue Book.  Issue 3.  Washington, D.C.: CCSDS, January 2002).

END
```

A2.7 SLE TRANSFER SERVICE—RAF OUTGOING PDUS

```
CCSDS-SLE-TRANSFER-SERVICE-RAF-OUTGOING-PDUS
{iso identified-organization(3) standards-producing-organization(112)
    ccsds(4) space-link-extension(3) sle-transfer-services(1)
    modules(1) return-all-frames-service(11) version-one(1)
    asn1-outgoing-pdu(3)}

DEFINITIONS
IMPLICIT TAGS
::= BEGIN

IMPORTS     Credentials
,           IntUnsignedLong
,           InvokeId
,           SpaceLinkDataUnit
,           Time
    FROM CCSDS-SLE-TRANSFER-SERVICE-COMMON-TYPES
            SleAcknowledgement
,           SleScheduleStatusReportReturn
    FROM CCSDS-SLE-TRANSFER-SERVICE-COMMON-PDUS
            AntennaId
,           CarrierLockStatus
,           DiagnosticRafGet
,           DiagnosticRafStart
,           FrameQuality
,           FrameSyncLockStatus
,           LockStatus
,           Notification
,           RafGetParameter
,           RafProductionStatus
,           SymbolLockStatus
    FROM CCSDS-SLE-TRANSFER-SERVICE-RAF-STRUCTURES
            SleBindInvocation
,           SleBindReturn
,           SlePeerAbort
,           SleUnbindInvocation
,           SleUnbindReturn
    FROM CCSDS-SLE-TRANSFER-SERVICE-BIND-TYPES
;

-- ============================================================
-- The first part of the module definition contains the RAF type
-- that contains all the possible PDUs the provider may send.
-- ============================================================

RafProviderToUserPdu        ::= CHOICE
{ rafBindInvocation                 [100]   SleBindInvocation
, rafBindReturn                     [101]   SleBindReturn
, rafUnbindInvocation               [102]   SleUnbindInvocation
, rafUnbindReturn                   [103]   SleUnbindReturn
, rafStartReturn                    [1]     RafStartReturn
, rafStopReturn                     [3]     SleAcknowledgement
, rafTransferBuffer                 [8]     RafTransferBuffer
, rafScheduleStatusReportReturn     [5]     SleScheduleStatusReportReturn
, rafStatusReportInvocation         [9]     RafStatusReportInvocation
, rafGetParameterReturn             [7]     RafGetParameterReturn
, rafPeerAbortInvocation            [104]   SlePeerAbort
}
```

```
-- ============================================================
-- The second part of the module definition contains the types
-- used by the RAF-PDUs declared in the first part.
-- ============================================================

FrameOrNotification             ::= CHOICE
{   annotatedFrame                      [0] RafTransferDataInvocation
,   syncNotification                    [1] RafSyncNotifyInvocation
}

RafGetParameterReturn           ::= SEQUENCE
{   performerCredentials                Credentials
,   invokeId                            InvokeId
,   result                              CHOICE
    {   positiveResult                      [0] RafGetParameter
    ,   negativeResult                      [1] DiagnosticRafGet
    }
}

RafStartReturn                  ::= SEQUENCE
{   performerCredentials                Credentials
,   invokeId                            InvokeId
,   result                              CHOICE
    {   positiveResult                      [0] NULL
    ,   negativeResult                      [1] DiagnosticRafStart
    }
}

RafStatusReportInvocation       ::= SEQUENCE
{   invokerCredentials                  Credentials
,   errorFreeFrameNumber                IntUnsignedLong
,   deliveredFrameNumber                IntUnsignedLong
,   frameSyncLockStatus                 FrameSyncLockStatus
,   symbolSyncLockStatus                SymbolLockStatus
,   subcarrierLockStatus                LockStatus
,   carrierLockStatus                   CarrierLockStatus
,   productionStatus                    RafProductionStatus
}

RafTransferBuffer               ::= SEQUENCE OF FrameOrNotification

RafSyncNotifyInvocation         ::= SEQUENCE
{   invokerCredentials                  Credentials
,   notification                        Notification
}

RafTransferDataInvocation   ::= SEQUENCE
{   invokerCredentials              Credentials
,   earthReceiveTime                Time
,   antennaId                       AntennaId
,   dataLinkContinuity              INTEGER (-1 .. 65535167772l5)
,   deliveredFrameQuality           FrameQuality
,   privateAnnotation               CHOICE
    {   null                            [0] NULL
    ,   notNull                         [1] OCTET STRING (SIZE (1 .. 128))
    }
,   data                            SpaceLinkDataUnit
}

END
```

ANNEX C

INDEX TO DEFINITIONS

(INFORMATIVE)

This annex lists terms used in this Recommended Standard and, for each term, provides a reference to the definition of that term.

Term	Reference
(data) type	reference [7]
(data) value	reference [7]
abstract binding	reference [1]
abstract object	reference [1]
abstract port	reference [1]
abstract service	reference [1]
abstract syntax	reference [6]
Abstract Syntax Notation One (ASN.1)	reference [7]
active (state)	subsection 2.6.4.2
AOS Transfer Frame	reference [4]
application entity	reference [6]
application layer	reference [6]
association	subsection 1.6.1.7.1
attached sync marker	reference [2]
codeblock	reference [2]
communications service	subsection 1.6.1.7.2
complete (online delivery mode)	subsections 2.3, 3.1.9.1.2
concatenation	reference [6]
confirmed operation	subsection 1.6.1.7.3
convolutional code	reference [2]
Cyclic Redundancy Check (CRC)	reference [4]
delivery criteria	subsection 1.6.1.7.4
~~error control field~~	~~1.6.1.8.5, 1.6.1.8.6~~
flow control	reference [6]
~~f~~Frame ~~e~~Error ~~c~~Control ~~f~~Field	reference~~s~~ [3], [4]
initiator	subsection 1.6.1.7.6
invocation	subsection 1.6.1.7.7

Term	Reference
SLE data channel	reference [1]
SLE Functional Group (SLE-FG)	reference [1]
SLE Protocol Data Unit (SLE-PDU)	reference [1]
SLE Service Data Unit (SLE-SDU)	reference [1]
SLE service package	reference [1]
~~SLE System~~	~~reference [1]~~
SLE transfer service instance	reference [1]
SLE transfer service production	reference [1]
SLE transfer service provision	reference [1]
SLE Utilization Management	reference [1]
space link	reference [1]
space link data channel	reference [1]
Space Link Data Unit (SL-DU)	reference [1]
space link session	reference [1]
telemetry frame	subsection 1.6.1.7.15
timely (online delivery mode)	subsections 2.3, 3.1.9.1
transfer buffer	subsections 2.6.4.6.2, 3.1.9
transfer frame	reference [3]
unbound (state)	subsection 2.6.4.2
unconfirmed operation	subsection 1.6.1.7.17
user-initiated	subsections 2.3, 3.2.1.2

ANNEX E

INFORMATIVE REFERENCES

(INFORMATIVE)

[E1] *Procedures Manual for the Consultative Committee for Space Data Systems*. CCSDS A00.0-Y-9. Yellow Book. Issue 9. Washington, D.C.: CCSDS, November 2003.

[E2] *Cross Support Concept — Part 1: Space Link Extension Services*. Report Concerning Space Data System Standards, CCSDS 910.3-G-3. Green Book. Issue 3. Washington, D.C.: CCSDS, March 2006.

[E3] *Telemetry Channel Coding*. Recommendation for Space Data System Standards, CCSDS 101.0-B-6-S. Historical Recommendation. Issue 6-S. Washington, D.C.: CCSDS, (October 2002) August 2005.

[E4] *Packet Telemetry*. Recommendation for Space Data System Standards, CCSDS 102.0-B-5-S. Historical Recommendation. Issue 5-S. Washington, D.C.: CCSDS, (November 2000) August 2005.

[E5] *Advanced Orbiting Systems, Networks and Data Links: Architectural Specification*. Recommendation for Space Data System Standards, CCSDS 701.0-B-3-S. Historical Recommendation. Issue 3-S. Washington, D.C.: CCSDS, (June 2001) August 2005.

[E6] *Space Link Extension—Internet Protocol for Transfer Services*. Recommendation for Space Data System Standards, CCSDS 913.1-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, September 2008.